

# CONTENT BASED MOVIE RECOMMENDING SYSTEM

D. Dilip Reddy<sup>1</sup>, Diddi Venkatesh<sup>2</sup>, Sandeep Lonkala<sup>3</sup>

<sup>1</sup> Student, Information Technology, B V Raju Institute of Technology, Telangana, India

<sup>2</sup> student, Information Technology, B V Raju Institute of Technology, Telangana, India

<sup>3</sup> Associate Professor, Information Technology, B V Raju Institute of Technology, Telangana, India

## ABSTRACT

*In the current era, most people are habituated to watching movies. Different people aspire to watch different genres of movies. So, the primary goal of movie recommendation systems is to filter and predict only those movies that a corresponding user is most likely to watch. In this, a web application is built to recommend five movies respective to the given movie using a machine learning model which converts content data to vectors and finds the similarity between movies through the angle between each vector. In this project, we used a content-based movie recommendation system. Content-based systems use metadata such as genre, producer, actor, and musician to recommend items say movies or music. Such a recommendation would be for instance recommending Infinity War which featured Vin Diesel because someone watched and liked The Fate of the Furious. Similarly, you can get music recommendations from certain artists because you liked their music. Content-based systems are based on the idea that if you liked a certain item, you are most likely to like something like it.*

**Keyword:** Python, Streamlit, Libraries (ast, nltk, pickle).

---

## 1. INTRODUCTION

There are three techniques of recommendation system: Collaborative Filtering, Content-Based Filtering and Hybrid Filtering. In Content Based recommender system, user provides data either explicitly (rating) or implicitly (by clicking on a link). The system captures this data and generates user profile for every user. By making use of user profile, recommendation is generated. In content-based filtering, recommendation is given by only watching single user's profile. System tries to recommend item like that item based on users past activity. Unlike content based, collaborative filtering finds those users whose likings are like a given user.

It then recommends item or any product, by considering that the given user will also like the item which other users like because their taste is similar. Both these Technique have their own strength and weakness so to overcome this, hybrid technique came into picture, which is a combination of both these techniques. Hybrid filtering can be used in various types. We can use content-based filtering first and then pass those results to collaborative recommender (and vice-versa) or by integrating both the filter into one model to generate the result. These kinds of modifications are also uses to cope up with cold start, data sparsity and scalability problem. Taxonomy of Recommender-System.

## Content-Based Filtering

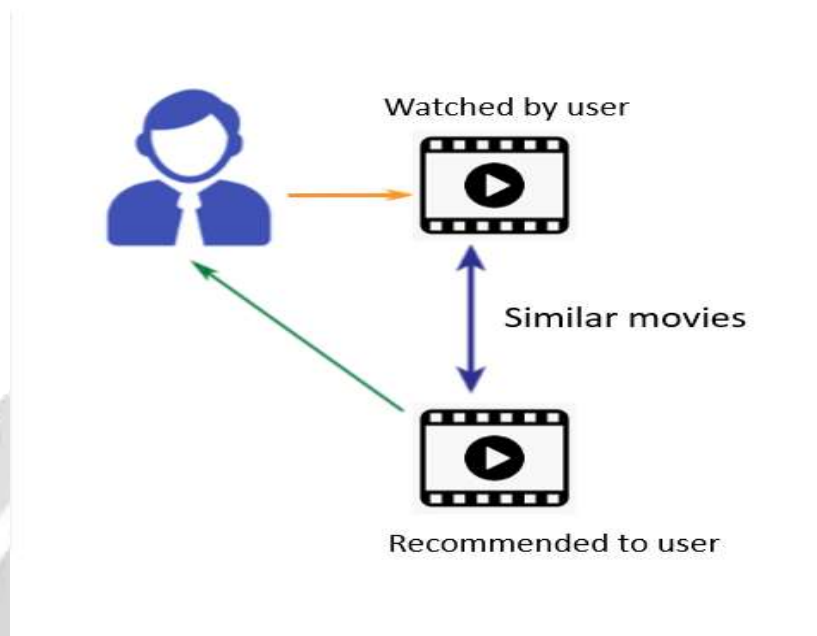


Fig. 1 Displays the how the movies are recommended to the users.

:

### 1.1 Cosine Similarity Algorithm

To find similar content for our item, we used the cosine similarity algorithm. The dot product between two vectors is equal to the projection of one of them on the other. Therefore, the dot product of two identical vectors is equal to their squared modules. On the other hand, if the two vectors do not share any directions, the product will be zero. General formula for calculating dot product is given below. This dot product is important when defining the similarity as it is directly connected to it. The definition of similarity between two vectors  $u$  and  $v$  is in fact the ratio between their dot products and product of their magnitudes. Thus, this will be equal to 1 if the two vectors are identical or it will be 0 if the two are orthogonal.

### 1.2 Bag Of Words Algorithm

The bag-of-words model is commonly used in methods of document classification where the (frequency of) occurrence of each word is used as a feature for training a classifier

## 2. METHODOLOGY

Content-Based Movie Recommender Systems are designed to suggest movies to users based on the characteristics of the movies they have enjoyed in the past. This type of recommender system utilizes the content of the movies to generate personalized recommendations for the users. Here is a detailed methodology of a Content-Based Movie Recommender System:

### Data Collection:

The first step in building a Content-Based Movie Recommender System is to collect data about movies and users. Various sources such as IMDb, Rotten Tomatoes, and Movie Lens can provide data about movie titles, genres, actors, directors, ratings, and plot summaries. The data collection process should also include information about the users, such as their ratings and reviews of the movies they have watched.

### Data Pre-processing:

After collecting the data, the next step is to pre-process it to remove inconsistencies, errors, or missing values. Also,

convert categorical data such as genres and actors into numerical data. Data pre-processing is crucial to ensure the accuracy and reliability of the recommender system.

**Feature Extraction:**

The next step is to extract features from the pre-processed data. Features can be any relevant attributes of the movies, such as genres, actors, directors, or any other descriptive characteristics that might influence a user's decision to watch a movie.

**Feature Vectorization:**

After extracting the features, the next step is to convert them into a feature vector for each movie. A feature vector can be a one-hot encoding or a binary vector that represents the presence or absence of the feature in the movie.

**Similarity Calculation:**

The feature vectors are then used to calculate the similarity between the movies. Similarity can be calculated using various similarity metrics such as cosine similarity or Euclidean distance. The similarity calculation helps to identify movies that are like each other based on their features.

**Recommendation Generation:**

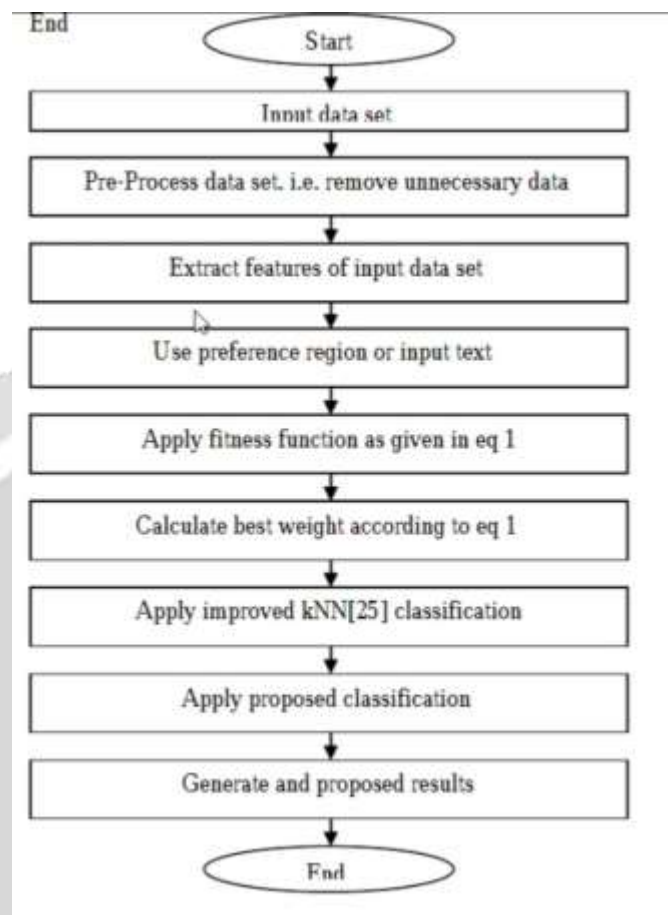
Once the similarity between the movies is calculated, the next step is to generate recommendations for a user based on the movies they have already watched and enjoyed. The recommender system searches for movies that are like the movies that the user has already watched and enjoyed. The system then recommends these similar movies to the user.

**Evaluation:**

The performance of the recommender system can be evaluated using various metrics such as precision, recall, and F1-score. These metrics measure the accuracy and effectiveness of the system in generating personalized recommendations for users.

**Deployment:**

Finally, the Content-Based Movie Recommender System can be deployed on a platform where users can access it and receive movie recommendations. The deployment of the recommender system can include user interfaces that allow users to rate movies and provide feedback on the recommendations.



**Fig:** Architecture of Proposed System

## 2.1 DESCRIPTION OF MODULES

Modules present in this Content Based Movie Recommending System project are:

1. **User module:** Takes the input from the users the attributes considered for Recommending the Movies.
2. **Prediction module:** Responsible for predicting Best Five Movies from the dataset.

### 3. RESULTS

The proposed system's results are shown in the figures below.

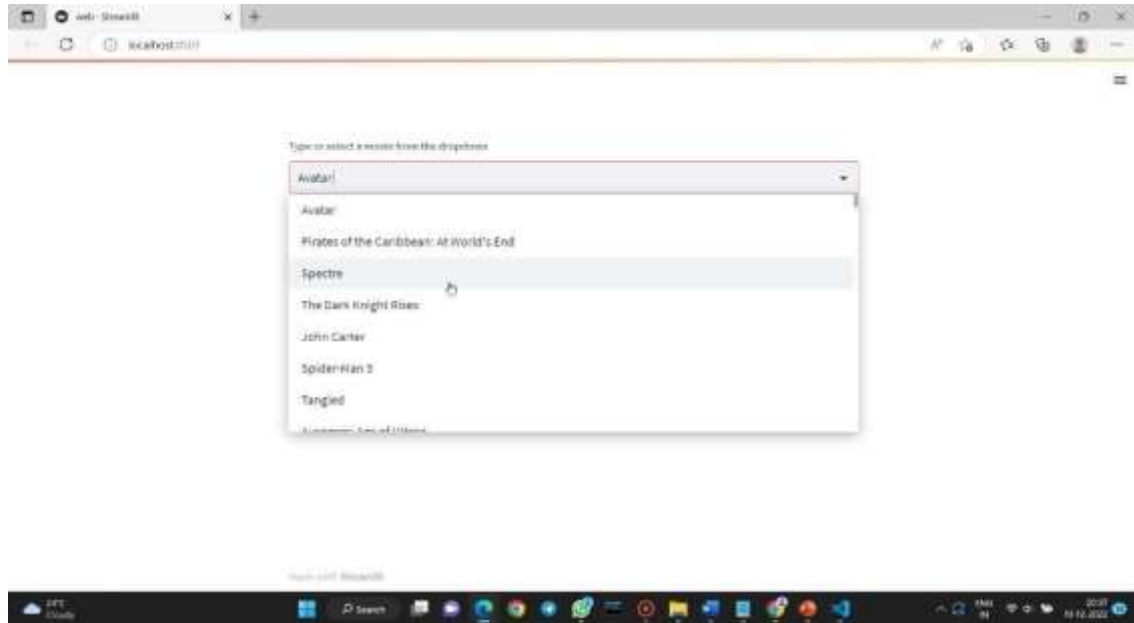


Fig: List of movies

The proposed system results are shown below in a diagrammatical representation. It displays five movies that best matches the search movie.

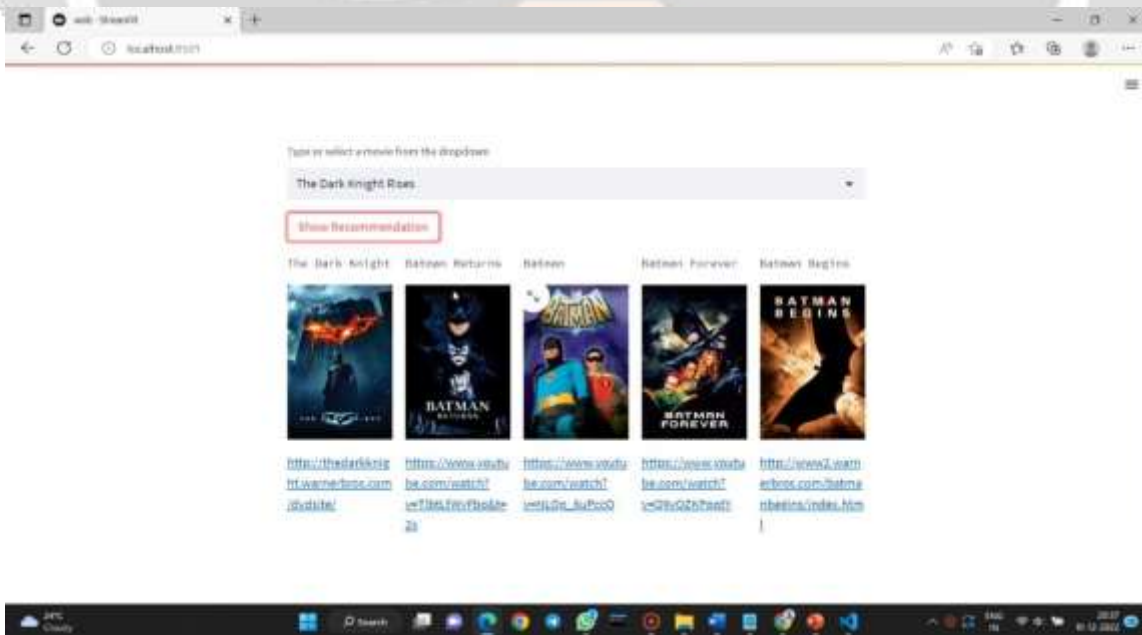


Fig: Displays the recommended movies.

#### 4. CONCLUSIONS

Content-Based Movie Recommender Systems offer a personalized way for users to discover new movies based on their personal preferences and tastes. These systems use the characteristics of the movies that the users have already enjoyed generating recommendations. The methodology of building a Content-Based Movie Recommender System involves data collection, pre-processing, feature extraction, feature vectorization, similarity calculation, recommendation generation, evaluation, and deployment.

By utilizing a Content-Based Movie Recommender System, users can receive movie recommendations that are tailored to their individual interests and preferences. This approach can improve the user experience by reducing the time and effort required to search for new movies to watch. Additionally, Content-Based Movie Recommender Systems can benefit movie streaming services by increasing user engagement and retention.

Overall, Content-Based Movie Recommender Systems are a powerful tool for personalized movie recommendation. As technology continues to evolve, these systems will continue to improve and offer even more accurate and effective movie recommendations to users.

#### 5. REFERENCES

- [1]. P. R. Mistry and P. M. Jadhav, "A Survey of Content-Based Recommender Systems for Movies," *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 5, pp. 40-47, May 2014.
- [2]. A. Chowdhury, R. Bose and S. Chakraborty, "Content-Based Movie Recommender System Using Deep Learning," *Proceedings of the International Conference on Computer Networks, Big Data and IoT*, vol. 1, pp. 227-233, August 2020.
- [3]. M. Verma, A. Bansal and G. K. Singh, "Content-Based Movie Recommender System Using A. Chowdhury, R. Bose and S. Chakraborty, "Content-Based Movie Recommender System Using Deep Learning," *Proceedings of the International Conference on Computer Networks, Big Data and IoT*, vol. 1, pp. 227-233, August 2020.
- [4]. H. K. Kim, J. H. Park and C. G. Kim, "A Content-Based Movie Recommender System Using User Preferences and Item Descriptions," *International Journal of Computer Science and Information Security*, vol. 14, no. 2, pp. 57-62, February 2016.
- [5]. M. J. Pazzani and D. Billsus, *Content-based recommendation systems*, in *The Adaptive Web*. Berlin, Germany: Springer, 2007, pp. 3253